

# Different Adiabatic Quantum Optimization Algorithms for the NP-Complete Exact Cover and 3SAT Problems

Vicky Choi

*vchoi@cs.vt.edu*

Department of Computer Science

Virginia Tech

Falls Church, VA

June 1, 2011

## Abstract

One of the most important questions in studying quantum computation is: whether a quantum computer can solve NP-complete problems more efficiently than a classical computer? In 2000, Farhi, et al. (Science, 292(5516):472–476, 2001) proposed the adiabatic quantum optimization (AQO), a paradigm that directly attacks NP-hard optimization problems. How powerful is AQO? Early on, van Dam and Vazirani claimed that AQO failed (i.e. would take exponential time) for a family of 3SAT instances they constructed. More recently, Altshuler, et al. (Proc Natl Acad Sci USA, 107(28): 12446–12450, 2010) claimed that AQO failed also for random instances of the NP-complete Exact Cover problem. In this paper, we make clear that all these negative results are only for a *specific* AQO algorithm. We do so by demonstrating different AQO algorithms for the same problem for which their arguments no longer hold. Whether AQO fails or succeeds for solving the NP-complete problems (either the worst case or the average case) requires further investigation. Our AQO algorithms for Exact Cover and 3SAT are based on the polynomial reductions to the NP-complete Maximum-weight Independent Set (MIS) problem.

## 1 Introduction

A quantum computer promises extraordinary power over a classical computer, as demonstrated by Shor [1] in 1994 with the polynomial quantum algorithm for solving the factoring problem, for which the best known classical algorithms are exponential. Just how much more powerful are quantum computers? In particular, can a quantum computer solve NP-complete problems [2] *more* efficiently than a classical computer? NP-complete problems are the “hardest” problems in NP — in the sense that if one can solve one NP-complete problem efficiently (i.e., in polynomial time) then one can solve all the problems in NP in polynomial time. In 2000, Farhi et al.[3, 4] proposed an adiabatic quantum algorithm as an alternative quantum paradigm to directly solve NP-hard optimization problems, which are polynomially equivalent to their corresponding NP-complete decision problems. Apparently, the same idea to the adiabatic quantum optimization, under a different name of *quantum annealing*, was first put forward by Apolloni et al. in 1988, see [5, 6] and references therein for a history of the field.

## 1.1 Adiabatic Quantum Algorithm

An adiabatic quantum algorithm is described by a time-dependent system Hamiltonian

$$\mathcal{H}(t) = (1 - s(t))\mathcal{H}_{\text{init}} + s(t)\mathcal{H}_{\text{problem}} \quad (1)$$

for  $t \in [0, T]$ ,  $s(0) = 0$ ,  $s(T) = 1$ . There are three components of  $\mathcal{H}(\cdot)$ : (1) initial Hamiltonian:  $\mathcal{H}(0) = \mathcal{H}_{\text{init}}$ ; (2) problem Hamiltonian:  $\mathcal{H}(T) = \mathcal{H}_{\text{problem}}$ ; and (3) evolution path:  $s : [0, T] \rightarrow [0, 1]$ , e.g.,  $s(t) = \frac{t}{T}$ .  $\mathcal{H}(t)$  is an adiabatic algorithm for a problem if we encode the problem into the problem Hamiltonian  $\mathcal{H}_{\text{problem}}$  such that the ground state of  $\mathcal{H}_{\text{problem}}$  corresponds to the answer to the problem. The initial Hamiltonian  $\mathcal{H}_{\text{init}}$  is chosen to be non-commutative with  $\mathcal{H}_{\text{problem}}$  and its ground state must be known and experimentally constructable, e.g.,  $\mathcal{H}_{\text{init}} = -\sum_{i \in V(G)} \Delta_i \sigma_i^x$ . Here  $T$  is the running time of the algorithm. According to the adiabatic theorem, if  $\mathcal{H}(t)$  evolves “slowly” enough, or equivalently, if  $T$  is “large” enough, which scales polynomially with the inverse of the *minimum spectral gap*  $g_{\min}$  (the difference between the two lowest energy levels) of the system Hamiltonian, the system remains in the ground state of  $\mathcal{H}(t)$ , and consequently, ground state of  $\mathcal{H}(T) = \mathcal{H}_{\text{problem}}$  gives the solution to the problem.

This computational model is referred as the *Adiabatic Quantum Computation* (AQC). It has been shown[7, 8] that AQC is polynomially equivalent to conventional quantum computation (quantum circuit model). For the optimization problem, the problem Hamiltonian can be expressed as a diagonal matrix in the computational basis. That is, let  $f_{\text{problem}} : \{0, 1\}^n \rightarrow \mathbb{R}$  be a cost function of the optimization problem such that the minimum of the  $f_{\text{problem}}$  corresponds to the solution of the optimization problem, then the corresponding problem Hamiltonian  $\mathcal{H}_{\text{problem}}$  is the Hamiltonian with  $f_{\text{problem}}$  as the energy function: namely,  $\mathcal{H}_{\text{problem}} = \sum_{x \in \{0, 1\}^n} f_{\text{problem}}(x) |x\rangle\langle x|$  (which needs to be expressible in polynomial resources, such as Eq. (4)). Hereafter we use the cost function and the energy function of the problem Hamiltonian interchangeably. It is worthwhile to emphasize here that given a problem, there can be many possible cost functions and thus many possible problem Hamiltonians for the same problem. This restricted model is referred as the *Adiabatic Quantum Optimization* (AQO) (which is no longer polynomially equivalent to the quantum circuit model). We remark that this distinction between AQC and AQO was not made by Altshuler et al. [9] (They referred both as AQO). In this paper, the focus is on this restricted model.

## 1.2 What Does “AQO Fails” Mean?

The NP-complete problems that were initially proposed for AQO by Farhi et al.[3, 4] were 3SAT and a special case of 3SAT — Exact Cover 3 (EC3). As an example, they proposed a clause-violation cost function as the energy function of the problem Hamiltonian. Namely,  $f_{\text{problem}}(x) = \text{number of clauses violated by the assignment } x$ . This cost function (with perhaps constant difference) has been adopted by almost all the other adiabatic quantum computation works. In particular, van Dam and Vazirani [10] claimed that AQO failed to solve a family of 3SAT instances by showing that the AQO algorithm with this specific clause-violation cost function had the exponentially small minimum spectral gap. See Discussion for more discussion on some other similar claims([11, 12]). Recently, Altshuler et al. [9] claimed that AQO failed for the average case of NP-complete problems by arguing this specific clause-violation cost function based AQO algorithm failed for random instances of EC3, because of the Anderson localization phenomenon. While the cost function proposed is a “natural” one, nevertheless, it is not the only possible one. Recall that, according to the formulation of an AQO algorithm, the requirement of the problem Hamiltonian is that the ground state corresponds to the solution. There can be other cost functions with the same minimum (solution), e.g., see the reduction below. That is, there are other problem Hamiltonians that have the same ground state but a different energy spectrum. Given a problem, there are three components

(initial Hamiltonian, problem Hamiltonian, and evolution path) that specify an AQO algorithm for the problem. A change in one component (e.g. problem Hamiltonian) will result in a different AQO algorithm for the same problem. So how to prove that AQO fails for a problem? When is it sufficient to argue for one specific cost function (and thus specific problem Hamiltonian) and generalize to all other possible problem Hamiltonians?

### 1.3 NP-Complete Reduction Based Problem Hamiltonian

In this paper, we make clear that the arguments in both van Dam and Vazirani [10] and Altshuler et al. [9] for their specific AQO algorithm do not generalize. We do so by concretely describing another problem Hamiltonian for the same problem to which their arguments no longer apply. The counter-arguments we provide are simple enough that researchers who are not familiar with AQC and/or do not fully understand the arguments in [9, 10] can easily follow. Our problem Hamiltonians are based on the polynomial reductions – NP-complete problems, by definition, can be polynomially reducible to each other – to the Maximum-weight Independent Set problem (see e.g. [2]), which is one of the well studied NP-complete problems. Perhaps, a more interesting and important question is: would the NP-complete reduction make a difference for the adiabatic complexity of the problem? Recall that the reduction requires only the solution to be preserved, i.e. there is a polynomial time algorithm that maps the solution to the reduced problem to the solution to the original problem and vice versa (see e.g. [13]). In other words, the reduction might only preserve the solution (i.e. the ground state) and alter the energy levels of the problem Hamiltonian. In [14], we demonstrate that even in small examples, the minimum spectral gap can be increased drastically when the excited energy levels are changed, due to the freedom in selecting parameters in the problem Hamiltonian. Thus, different reductions are possible, giving rise to different problem Hamiltonians, and thus different AQO algorithms, for the same problem. What are the time complexities of these different AQO algorithms? Unfortunately, at this point, the analytical analysis of these algorithms, which requires bounding the minimum spectral gap of the system Hamiltonian, remains challenging. Whether the time complexity of each of these AQO algorithms is polynomial or exponential remains open and requires further investigation.

## 2 Methods and Results

In the following, we first recall the NP-complete Maximum-weight Independent Set (MIS) problem, and a problem Hamiltonian for solving MIS. Then we describe a simple reduction from Exact Cover to MIS that results in a different problem Hamiltonian (and thus different AQO algorithm) for Exact Cover. We then make clear that the argument in Altshuler et al. [9] does not apply to this algorithm. Similarly, we recall the well-known reduction from 3SAT to MIS, and point out that the argument in [10] does not apply to the MIS-based AQO algorithm.

### 2.1 Maximum-weight Independent Set

Formally, the Maximum-Weight Independent Set problem is as follows:

**Input:** An undirected graph  $G(= (V(G), E(G)))$ , where each vertex  $i \in V(G) = \{1, \dots, n\}$  is weighted by a positive rational number  $c_i$

**Output:** A subset  $S \subseteq V(G)$  such that  $S$  is independent (i.e., for each  $i, j \in V(G)$ ,  $i \neq j$ ,  $ij \notin E(G)$ ) and the total *weight* of  $S$  ( $= \sum_{i \in S} c_i$ ) is maximized. Denote the optimal set by  $\text{mis}(G)$ .

This is referred as the *optimization* version of MIS. The corresponding *decision* problem is:

**Question:** Is there an independent subset  $S \subseteq V(G)$  such that the total weight is at least  $k$ ? (where the positive rational number  $k$  is the extra input parameter.)

Technically speaking, the decision version of MIS is NP-complete, and the optimization version of MIS is NP-hard. Since the decision problem and the optimization problem are polynomially transformable to each other, in the following, we will simply refer to the MIS problem, without explicitly mentioning which version (optimization or decision). In general, which version should be clear from the context.

One way to solve the MIS problem is by maximizing a quadratic binary function  $\mathcal{Y}$  (also known as pseudo-boolean function) defined in the following theorem (Theorem 5.1 in [15]).

**Theorem 2.1.** *If  $J_{ij} \geq \min\{c_i, c_j\}$  for all  $ij \in E(G)$ , then the maximum value of*

$$\mathcal{Y}(x_1, \dots, x_n) = \sum_{i \in V(G)} c_i x_i - \sum_{ij \in E(G)} J_{ij} x_i x_j \quad (2)$$

*is the total weight of the MIS. In particular if  $J_{ij} > \min\{c_i, c_j\}$  for all  $ij \in E(G)$ , then  $\text{mis}(G) = \{i \in V(G) : x_i^* = 1\}$ , where  $(x_1^*, \dots, x_n^*) = \arg \max_{(x_1, \dots, x_n) \in \{0,1\}^n} \mathcal{Y}(x_1, \dots, x_n)$ .*

Notice that in this formulation, we only require  $J_{ij} > \min\{c_i, c_j\}$ , and thus there is freedom in choosing this parameter. It is easy to see (by changing the variables  $x_i = \frac{1+s_i}{2}$ ) that MIS is equivalent to minimizing the *Ising energy function* :

$$\mathcal{E}(s_1, \dots, s_n) = \sum_{i \in V(G)} h_i s_i + \sum_{ij \in E(G)} J_{ij} s_i s_j, \quad (3)$$

which is the eigenfunction of the following *Ising Hamiltonian*:

$$\mathcal{H}_C = \sum_{i \in V(G)} h_i \sigma_i^z + \sum_{ij \in E(G)} J_{ij} \sigma_i^z \sigma_j^z \quad (4)$$

where  $h_i = \sum_{j \in \text{nbr}(i)} J_{ij} - 2c_i$ ,  $\text{nbr}(i) = \{j : ij \in E(G)\}$ , for  $i \in V(G)$ .

In other words, using the Ising Hamiltonian  $\mathcal{H}_C$  as the problem Hamiltonian, we have an AQO algorithm for solving the MIS problem. At the end of the evolution, the ground state of  $\mathcal{H}_C$ , say  $|x_1^* x_2^* \dots x_n^*\rangle$ , corresponds to the maximum-weight independent set, namely  $\text{mis}(G) = \{i : x_i^* = 0\}$ .

## 2.2 Exact Cover

Formally, the Exact Cover is as follows:

**Input:** A set of  $m$  elements,  $X = \{c_1, c_2, \dots, c_m\}$ ; a family of  $n$  subsets of  $X$ ,  $\mathcal{S} = \{S_1, S_2, \dots, S_n\}$ , where  $S_i \subset X$

**Question:** Is there a subset  $I \subseteq \{1, \dots, n\}$  such that  $\cup_{i \in I} S_i = X$ , where  $S_i \cap S_j = \emptyset$  for  $i \neq j \in I$ ? Here  $\{S_i : i \in I\}$  is called an *exact cover* of  $X$ .

See Figure 1 for an example. In particular, in this example, each element  $c_i \in X$  appears exactly in three subsets. This kind of special instance is referred as EC3, which can then be polynomially reducible ( $\leq_P$ ) to the positive 1-in-3SAT problem.

**EC3  $\leq_P$  positive 1-in-3SAT.** Given an instance of EC3 with an  $m$ -element set  $X$  and  $n$  subsets  $S_1, \dots, S_n$ , we construct a 3CNF boolean formula  $\Psi(x_1, \dots, x_n) = C_1 \wedge \dots \wedge C_m$  with  $n$  variables and  $m$  clauses. For each element  $c_i \in X$ , let  $S_{i_1}, S_{i_2}, S_{i_3}$  be the three sets that consist of  $c_i$ . By definition, exactly one of these three sets needs to be in the exact cover of  $X$ . Therefore, if we define a binary variable  $x_k$  for each set  $S_k$  such that  $x_k = 1$

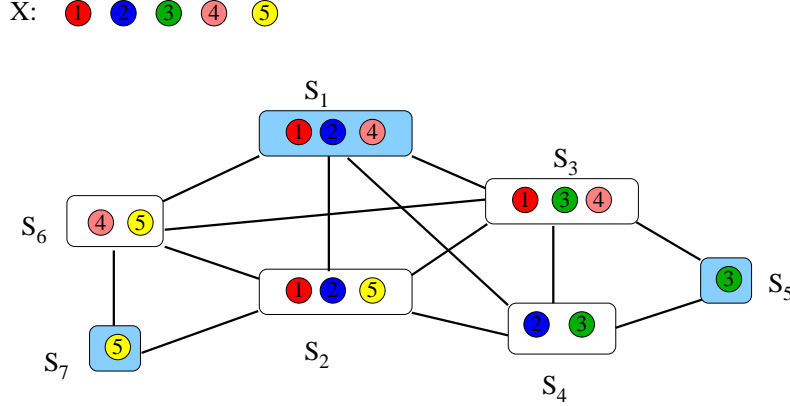


Figure 1:  $X = \{c_1, c_2, c_3, c_4, c_5\}$ , and  $\mathcal{S} = \{S_1, S_2, \dots, S_7\}$ , with  $S_1 = \{c_1, c_2, c_4\}$ ,  $S_2 = \{c_1, c_2, c_5\}$ ,  $S_3 = \{c_1, c_3, c_4\}$ ,  $S_4 = \{c_2, c_3\}$ ,  $S_5 = \{c_3\}$ ,  $S_6 = \{c_4, c_5\}$ ,  $S_7 = \{c_5\}$ . Here  $\{S_1, S_5, S_7\}$  is the exact cover of  $X$ . In this example, each element  $c_i$  appears in exactly three subsets. For instance, element  $c_1$  (the red ball) appears in  $S_1, S_2$  and  $S_3$ . (a) The problem instance can be viewed as a graph  $G_M$  with  $n$  vertices, where vertex  $i$  corresponds to the set  $S_i$ , and there is an edge between two vertices if and only if  $S_i$  and  $S_j$  overlap (share a common element). There is an exact cover of  $X$  if and only if the weight of  $\text{mis}(G_M)$  is  $m$ , where the weight of vertex  $i$  is the number of elements in  $S_i$ . (b) The problem can also be reduced to the following instance of positive 1-in-3SAT:  $\Psi(x_1, \dots, x_7) = C_1 \wedge C_2 \wedge C_3 \wedge C_4 \wedge C_5$  where  $C_1 = x_1 \vee x_2 \vee x_3$ ,  $C_2 = x_1 \vee x_2 \vee x_4$ ,  $C_3 = x_3 \vee x_4 \vee x_5$ ,  $C_4 = x_1 \vee x_3 \vee x_6$ ,  $C_5 = x_2 \vee x_6 \vee x_7$ . There is an exact cover of  $X$  if only if  $\Psi$  is satisfiable in that there is exactly one variable in each clause is true.

if  $S_k$  is in the exact cover, and  $x_k = 0$  otherwise, we require exactly one variable in  $C_i = x_{i_1} \vee x_{i_2} \vee x_{i_3}$  to be satisfied. That is, there is an exact cover to the original problem if and only if the formula  $\Psi(x_1, \dots, x_n) = C_1 \wedge \dots \wedge C_m$  is satisfiable in that there is exactly one variable in each clause is satisfied, which is the positive 1-in-3SAT problem. See Figure 1(b) for an example of the reduction.

One way of solving the positive 1-in-3SAT problem is to minimize the following clause-violation cost function

$$\mathcal{E}_\Psi(x_1, \dots, x_n) = \sum_{i=1}^m (x_{i_1} + x_{i_2} + x_{i_3} - 1)^2.$$

$\Psi$  is satisfiable if and only if the minimum of  $\mathcal{E}_\Psi$  is *zero* (i.e. no violation). The corresponding problem Hamiltonian based on this cost function as used by Altshuler et al.[9] (and Young et al. [16])<sup>1</sup> is

$$\mathcal{H}_A = \sum_{i \in V(G_{\text{EC}})} B_i \sigma_i^z + \sum_{ij \in E(G_{\text{EC}})} I_{ij} \sigma_i^z \sigma_j^z \quad (5)$$

where  $B_i$  is the number of clauses that contains variable  $x_i$ , and  $I_{ij}$  is the number of clauses that contains both  $x_i$  and  $x_j$ , and  $V(G_{\text{EC}}) = \{1, \dots, n\}$ , and  $E(G_{\text{EC}}) = \{ij : x_i \text{ and } x_j \text{ appear in a clause}\}$ .

Next, we show that there is another way of looking at the Exact Cover problem, which leads to a simple polynomial reduction from Exact Cover to MIS.

**Exact Cover  $\leq_P$  MIS.** Given an instance of Exact Cover with an  $m$ -element set  $X$  and  $n$  subsets  $S_1, \dots, S_n$ . Recall that each element in  $X$  can appear in exactly in one set. If two sets overlap, e.g.  $S_1$  and  $S_2$  in Figure 1,

<sup>1</sup>The sign of  $\sigma_i^z$  term is in opposite because they use  $x_i = \frac{1-s_i}{2}$  instead.  $I_{ij}$  was called  $J_{ij} (= \frac{1}{2}(J_{ij} + J_{ji}))$  in [9].

then they can not both appear in the exact cover. Therefore, if we construct a graph  $G_M$  with  $n$  vertices, where vertex  $i$  corresponds to the set  $S_i$ , and there is an edge between two vertices if  $S_i$  and  $S_j$  overlap. Then an exact cover of  $X$  will correspond to an independent set of  $G_M$ . Moreover, if we let the weight of vertex  $i$  be the number of elements in  $S_i$ , then there is an exact cover to the original problem if and only if the weight of  $\text{mis}(G_M)$  is  $m$ .

For EC3, it is easy to see that  $G_{\text{EC}}$  and  $G_M$  are exactly the same because there is one-one corresponding between the variable  $x_i$  and the set  $S_i$  ( $V(G_{\text{EC}}) = V(G_M)$ ), and “ $x_i$  and  $x_j$  appear in a clause” is equivalent to “ $S_i$  and  $S_j$  share a common element” ( $E(G_{\text{EC}}) = E(G_M)$ ). Based on this reduction, we therefore have the following problem Hamiltonian for the same problem:

$$\mathcal{H}_C = \sum_{i \in V(G_{\text{EC}})} \left( \sum_{j \in \text{nbr}(i)} J_{ij} - 2B_i \right) \sigma_i^z + \sum_{ij \in E(G_{\text{EC}})} J_{ij} \sigma_i^z \sigma_j^z \quad (6)$$

where  $J_{ij} > \min\{B_i, B_j\}$ .

**Comparison of  $\mathcal{H}_A$  and  $\mathcal{H}_C$ .** For comparison, recall that  $2B_i = \sum_{j \in \text{nbr}(i)} I_{ij}$ , let us write  $J_{ij} = 2I_{ij} + D_{ij}$ , where  $D_{ij} > \min\{B_i, B_j\} - 2I_{ij}$ . Then we have

$$\mathcal{H}_C = 2\mathcal{H}_A + \sum_{i \in V(G_{\text{EC}})} \sum_{j \in \text{nbr}(i)} D_{ij} \sigma_i^z + \sum_{ij \in E(G_{\text{EC}})} D_{ij} \sigma_i^z \sigma_j^z.$$

Alternatively, as we show in [17], for  $D_{ij} > 0$ , we can view  $\mathcal{H}_C$  as obtained from the following clause-violation cost function:

$$\mathcal{E}'_{\Psi}(x_1, \dots, x_n) = \sum_{k=1}^m (x_{k_1} + x_{k_2} + x_{k_3} - 1)^2 + \sum_{ij \in E(G_{\text{EC}})} D_{ij} x_i x_j \quad (7)$$

$$= \mathcal{E}_{\Psi}(x_1, \dots, x_n) + \sum_{ij \in E(G_{\text{EC}})} D_{ij} x_i x_j \quad (8)$$

Notice that the extra term ( $\sum_{ij \in E(G_{\text{EC}})} D_{ij} x_i x_j$ ) will not contribute to the energy function for the truth assignment because at least one of  $\{x_i, x_j\}$  (for  $ij \in E(G_{\text{EC}})$ ) will be zero. In other words, both  $\mathcal{E}_{\Psi}$  and  $\mathcal{E}'_{\Psi}$  are the cost functions for the same problem instance  $\Psi$ , where  $D_{ij} > 0$  can be arbitrary. The question is: does this extra term (for arbitrarily chosen  $D_{ij} > 0$ ) matter to the performance of the adiabatic algorithm?

In [9], Altshuler et al. claimed that the AQO algorithm with problem Hamiltonian  $\mathcal{H}_A$  failed with high probability for randomly generated instances of EC3 due to the Anderson localization(AL). The authors claimed that the correctness of their argument did not rely on the specific form of the problem Hamiltonian for EC3, but only depended on the properties of the problem instance  $B_i$  and  $I_{ij}$ . However, our problem Hamiltonian  $\mathcal{H}_C$  belies their claim. Their argument necessarily depends on the energy function of the problem Hamiltonian. While the energy function for  $\mathcal{H}_A$  only depends on  $B_i$  and  $I_{ij}$ , the energy function for  $\mathcal{H}_C$  also depends on the extra  $D_{ij}$  whose values have a range to choose. Therefore, their argument which depends on the random property of parameters does not apply to the algorithm based on  $\mathcal{H}_C$  because  $D_{ij}$  are not random. Perhaps, their main message is that from the physics point of view, if AL occurs, then the AQO algorithm will require exponential time, as shown by Amin and Choi in [19]. So what are the necessary conditions for AL to occur? Can the arguments in [9] be modified such that AL still occurs, and the AQO algorithm with  $\mathcal{H}_C$  for arbitrarily chosen  $D_{ij}$  will also fail? Here we emphasize that while the connectivity of the graph is random, the values of  $D_{ij}$  (at least  $\min\{B_i, B_j\} - 2I_{ij}$ ) are free to choose. To show that the AQO algorithm with  $\mathcal{H}_C$  fails, one would need

to show that the algorithms correspond to *all* possible values of  $D_{ij}$  *all* fail. Or alternatively, can one show that there does not exist  $D_{ij}$  (not necessarily efficiently computable) such that the corresponding adiabatic algorithm is not at least exponential? In [16], Young et al. used QMC to show that the minimum spectral gap  $g_{\min}$  of the AQO algorithm (with the same problem Hamiltonian  $\mathcal{H}_A$ ) is exponentially small. It will be interesting to see the  $g_{\min}$  result (for the same set of instances) using this new problem Hamiltonian  $\mathcal{H}_C$  for some  $D_{ij}$ .

### 2.3 3SAT

Similarly, for 3SAT, there is a well-known reduction to MIS (which is one of the first NP-complete reductions, to show the NP-hardness of MIS) [2]. For completeness, here we recall the reduction:

**3SAT  $\leq_P$  MIS.** Given a 3SAT instance  $\Psi(x_1, \dots, x_n) = C_1 \wedge \dots \wedge C_m$  with  $n$  variables and  $m$  clauses, we construct a (unweighted) graph  $G_{\text{SAT}}$  as follows:

- For each clause  $C_i = y_{i_1} \vee y_{i_2} \vee y_{i_3}$ , we construct a triangle with three vertices labeled accordingly, i.e., with  $y_{i_1}, y_{i_2}, y_{i_3}$ , where  $y_j \in \{x_j, \bar{x}_j\}$ . Therefore,  $G_{\text{SAT}}$  consists of  $3m$  vertices.
- There is an edge between two vertices in different triangles if there labels are in conflict. That is, for  $i \neq j$ ,  $i_s j_t \in E(G_{\text{SAT}})$  if and only if  $y_{i_s} = \bar{y}_{j_t}$ .

One can then show that  $\Psi$  is satisfiable if and only if  $G_{\text{SAT}}$  has a MIS of size  $m$ . See Figure 2 for an example.

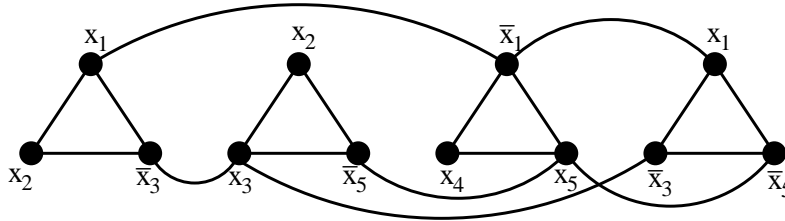


Figure 2: Graph  $G_{\text{SAT}}$  corresponds to  $\Psi(x_1, \dots, x_5) = (x_1 \vee x_2 \vee \bar{x}_3) \wedge (x_2 \vee x_3 \vee \bar{x}_5) \wedge (\bar{x}_1 \vee x_4 \vee x_5) \wedge (x_1 \vee \bar{x}_3 \vee \bar{x}_5)$ .  $\Psi$  is satisfiable if and only if  $G_{\text{SAT}}$  has a MIS of size 4.

In [10], van Dam and Vazirani constructed a special set of 3SAT instances. They showed that the clause-violation cost function based AQO algorithm would take exponential time for these instances because of the exponential small  $g_{\min}$ . Farhi et al.[18] showed that the exponential small gap could be overcome by different initial Hamiltonians. Here we point out also that the argument in [10] does not apply to the MIS based problem Hamiltonian. In order to show that the exponential small gap, their argument requires the cost function to be “ $\epsilon$ -deceptive monotone”. For completeness, here we recall the definitions:

**Definition 2.2.** (Definition 1 in [10]) Consider the hypercube  $\{0, 1\}^n$  with the partial ordering on its strings  $x \preceq y$  if and only if  $x_i \leq y_i$  for all  $i = 1, \dots, n$ . A function  $f : \{0, 1\}^n \rightarrow \mathbb{R}$  is monotonically decreasing if  $x \preceq y$  implies  $f(x) \geq f(y)$ . Similarly for monotonically increasing functions.

**Definition 2.3.** (Definition 2 in [10]) A cost function  $f : \{0, 1\}^n \rightarrow \mathbb{R}$  is “ $\epsilon$ -deceptive monotone” if it is monotone for all but a fraction  $\epsilon$  vertices comprising the top layers of the hypercube.

However, it is easy to see that the cost function (i.e. the pseudo-boolean function  $\mathcal{V}$  in Theorem 2.1) for MIS is not monotone. For example,  $\mathcal{V}(1, 0, 0, \dots, 0) > \mathcal{V}(1, 1, 0, \dots, 0)$  where vertex 1 and vertex 2 are adjacent, while

$\mathcal{V}(1, 0, 0, \dots, 0) < \mathcal{V}(1, 0, 0, \dots, 1)$  where vertex 1 and vertex  $n$  are independent. Therefore their argument no longer applies to this cost function, and one can no longer conclude that this AQO algorithm also fails for the instances they constructed.

### 3 Discussion

AQO was originally proposed in [3, 4] as an energy minimization algorithm that aims to use coherent quantum evolution to avoid trapping in the local minima that trip classical algorithms of NP-hard optimization problems. Early on, van Dam et al. [10, 11] and Reichardt [12] showed that the AQO algorithm failed to avoid local minima and would take exponential time for computing the minimum of some cost function  $f$ , where their AQO algorithm is defined such that  $f$  is the energy function of the problem Hamiltonian. In other words, in their formulation, the energy function of the problem Hamiltonian of their AQO algorithm is same as the cost function of the problem. However, given a problem, there are in general many possible cost functions with the same minimum. Thus, their AQO algorithm is just a *specific* algorithm of our more general AQO definition. This perhaps rather obvious difference unfortunately is not clear to even quantum computation experts who are not working on the adiabatic quantum computation, and the results of van Dam et al.[10, 11] and Reichardt[12] were widely believed that they held for all AQO algorithms. Technically speaking, the lower bound of this specific AQO algorithm for computing the minimum of  $f$  does not constitute a lower bound for all AQO algorithms of the same problem. Nevertheless, do these results provide “convincing evidence” that AQO would fail to solve problems with many local minima? For this purpose, we constructed a family of MIS graphs in which there are exponentially many local minima, our initial results, which were explained by the first order quantum phase transition [19], agreed with the speculation – that is, the system got trapped in the local minima and the particular AQO algorithm failed. However, in [14], we showed that the exponential small gap (caused by the system getting trapped in the local minima) could be overcome by changing the parameters of the problem Hamiltonian without changing the problem (that is, the global minimum and the exponential local minima remain the same). Although the result is only numerical and supported by visualization, this small example serves to clarify that it is not sufficient to consider one specific problem Hamiltonian (and thus specific AQO algorithm) for proving the adiabatic lower bound of a problem.

After the appearance of the arXiv version of this paper, F. Krzakala pointed out to the author that the failure of “specific” AQO algorithms for the random instances of some NP-complete problems was discussed in their work [20, 21]. Note that also, some initial Hamiltonian, such as the projection Hamiltonian used in [22], can make the corresponding AQO algorithm fail as shown in [23]. Recently, it was also shown in [24] that the lack of structure in the cost function of the problem Hamiltonian can make the corresponding AQO fail. Farhi et al. [25] suggested that an adiabatic quantum algorithm should be run on each instance with many different random paths (as an integral part of AQO).

In this paper, we describe different AQO algorithms for the NP-complete Exact Cover and 3SAT problems. These algorithms are rather straightforward, from the computer science point of view, as they are based on the simple NP-complete reductions. However, they serve to further clarify the distinction between one specific problem Hamiltonian AQO algorithm and general AQO algorithms for a problem. In particular, we make clear that the arguments in both van Dam and Vazirani [10] and Altshuler et al. [9] for their specific AQO algorithm do not apply to the AQO algorithms we describe here. While from the physics point of view, “Anderson Localization” makes an AQO algorithm fail, does AL necessary occur for all AQO algorithms of the same problem? What are the assumptions or conditions for AL to occur? In particular, can one modify the arguments in Altshuler et al. [9] such that the AQO algorithm described here also fails? Whether the time complexity of these different AQO algorithms is exponential or not requires more rigorous analysis. Our counter-arguments are so simple that one

does not need to fully understand the arguments in [9, 10] nor to have advanced algorithmic knowledge in order to understand. This perhaps highlights the challenge of this interdisciplinary research, and calls for more serious investigation.

## Acknowledgements

I would like to thank Peter Young and Boris Altshuler for the discussion. I would like to thank Siyuan Han, Mohammad Amin, Neil Dickson, Robert Raussendorf, Tzu-Chieh Wei and Pradeep Kiran for their comments. I would like to thank my very enthusiastic students in my adiabatic quantum computing class: Ryan Blace, Russell Brasser, Mark Everline, Eric Franklin, Nabil Al Ramli, and Aiman Shabsigh. Thanks also go to David Sankoff and David Kirkpatrick for the encouragement.

## References

- [1] P.W. Shor (1994;1997), *Algorithms for quantum computation: discrete logs and factoring*, Proc. 35th FOCS; SIAM J. Comp., 26, 1484–1509.
- [2] M. R. Garey and D. S. Johnson (1979), *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman.
- [3] E. Farhi, J. Goldstone, S. Gutmann, and M. Sipser (2000), *Quantum computation by adiabatic evolution*, arXiv:quant-ph/0001106.
- [4] E. Farhi, J. Goldstone, S. Gutmann, J. Lapan, A. Lundgren, and D. Preda (2001), *A quantum adiabatic evolution algorithm applied to random instances of an NP-complete problem*, Science, 292(5516):472–476.
- [5] G.E. Santoro and E. Tosatti (2006), *Optimization using quantum mechanics: quantum annealing through adiabatic evolution*, J. Phys. A: Math. Gen., 39, 393–431.
- [6] G.E. Santoro and E. Tosatti (2008), *Corrigendum: Optimization using quantum mechanics: quantum annealing through adiabatic evolution*, J. Phys. A: Math. Theor., 41.
- [7] D. Aharonov, W. van Dam, J. Kempe, Z. Landau, S. Lloyd, and O. Regev (2004;2007), *Adiabatic quantum computation is equivalent to standard quantum computation*, Proc. 45th FOCS; SIAM Journal of Computing, Vol. 37, Issue 1, p. 166–194.
- [8] A. Mizel, D.A. Lidar, and M. Mitchell (2007), *Simple Proof of Equivalence Between Adiabatic Quantum Computation and the Circuit Model*, Phys. Rev. Lett., 99, 070502.
- [9] B. Altshuler, H. Krovi and J. Roland (2010), *Anderson localization makes adiabatic quantum optimization fail*, Proc Natl Acad Sci USA, 107(28): 12446–12450.
- [10] W. van Dam and U. Vazirani (2001), *Limits on quantum adiabatic optimization*, Available at <http://www.cs.berkeley.edu/~vazirani/pubs/qao.ps>.
- [11] W. van Dam, M. Mosca, and U. Vazirani (2001), *How powerful is adiabatic quantum computation?*, Proc. 42nd FOCS, 279–287.

- [12] B.W. Reichardt (2004), *The quantum adiabatic optimization algorithm and local minima*, Proc. 35th STOC, 502–510.
- [13] S. Dasgupta, C. Papadimitriou, U. Vazirani (2008), *Algorithms*, McGraw Hill.
- [14] V. Choi (2010), *Avoid first order quantum phase transition by changing problem hamiltonians*, ArXiv:quant-ph/1010.1220.
- [15] V. Choi (2008), *Minor-embedding in adiabatic quantum computation: I. The parameter setting problem*, Quantum Inf. Processing, 7, 193–209.
- [16] A. P. Young and S. Knysh and V. N. Smelyanskiy (2009), *First order phase transition in the quantum adiabatic algorithm*, Phys. Rev. Lett., 104, 020502.
- [17] V. Choi (2011), *Different adiabatic quantum optimization algorithms for the NP-complete exact cover problem*, Proc Natl Acad Sci USA, 108(7): E19-E20.
- [18] E. Farhi, J. Goldstone and S. Gutmann (2002), *Quantum adiabatic evolution algorithms with different paths*, arXiv.org:quant-ph/0208135.
- [19] M.H.S. Amin, V. Choi (2009), *First order phase transition in adiabatic quantum computation*, Phys. Rev. A, 80, 062326.
- [20] T. Jorg, F. Krzakala, J. Kurchan and A. C. Maggs (2008), *Simple glass models and their quantum annealing*, Phys. Rev. Lett., 101, 147204.
- [21] T. Jorg, F. Krzakala, G. Semerjian and F. Zamponi (2010), *First-order transitions and the performance of quantum algorithms in random optimization problems*, Phys. Rev. Lett., 104, 207206.
- [22] Marko Znidaric and Martin Horvat (2006), *Exponential complexity of an adiabatic algorithm for an NP-complete problem*, Phys. Rev. A, 73, 022329.
- [23] E. Farhi, J. Goldstone, S. Gutmann, and D. Nagaj (2005), *How to make the quantum adiabatic algorithm fail*, International Journal of Quantum Information, 6(3):503–516.
- [24] E. Farhi, J. Goldstone, D. Gosset, S. Gutmann, P. Shor (2009), *Unstructured randomness, small gaps and localization*, arXiv:quant-ph/1010.0009.
- [25] E. Farhi, J. Goldstone, D. Gosset, S. Gutmann, H. B. Meyer and P. Shor (2011), *Quantum adiabatic algorithms, small gaps, and different paths*, Quantum Information and Computation, 11(3), 0181–0214.